UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/782,734	02/19/2004	Shafiq Ur Rahman	MS1-1848US	3409

22801 7590 11/16/2009

LEE & HAYES, PLLC
601 W. RIVERSIDE AVENUE
SUITE 1400
SPOKANE, WA 99201

EXAMINER	
ITURRALDE, ENRIQUE W	

ART UNIT	PAPER NUMBER
2179	

NOTIFICATION DATE	DELIVERY MODE
11/16/2009	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

lhptoms@leehayes.com

Office Action Summary	Application No.	Applicant(s)	
	10/782,734	RAHMAN ET AL.	
	Examiner	Art Unit	
	ENRIQUE W. ITURRALDE	2179	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *02 July 2009*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *See Continuation Sheet* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,2,4,5,7-9,11-18,20-26,28,30-35,37-41,44,45,47-62,64-68,70-79,82-87,89 and 90* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

Continuation of Disposition of Claims: Claims pending in the application are 1,2,4,5,7-9,11-18,20-26,28,30-35,37-41,44,45,47-62,64-68,70-79,82-87,89 and 90.

DETAILED ACTION

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

Claims 33-35, 37-40, 49-53 and 55-57 are rejected under 35 U.S.C. 102(a) as being anticipated by Deutscher (US 2004/0001106).

As per claim 33, Deutscher teaches in a media timeline exposed via an application programming interface and having a plurality of nodes, a method comprising: rendering [rendering of 0027-0028] a first media item referenced by a first node [tree and nodes of 0011, 0023, 0163-0166 and Figure 19]; receiving a call for a second node that references a second media item [call for items of 0100 of timeline of 0011 and 0026]; and creating the second node while rendering the first media item [rendering presentation timeline of 0027-0028]; and wherein the media timeline is configured for dynamic updating such that metadata included in at least one node specifies a collection of nodes to be modified when the at least one node is loaded [metadata for dynamic changes to output during rendering of 0213-0221 and 0230].

As per claim 34, Deutscher teaches: rendering a second media item referenced by the second node when the rendering of the first media item is completed [rendering presentation timeline of 0027-0028].

As per claim 35, Deutscher teaches: rendering the second media item referenced

by the second node; receiving a call for a third node that references a third media item; and creating the third node [rendering of 0027-0028, calling for items of a timeline of 0100].

As per claim 37, Deutscher teaches: at least one node is configured to reference an effect to be applied to an output of media referenced by the node [script commands and events of 0018].

As per claim 38, Deutscher teaches: at least one node is specified as read-only [cannot be edited of 0017].

As per claim 39, Deutscher teaches: at least one said node is configured for communication of events to another said node such that a change may be made to the media timeline while the media timeline is rendered [changes to an event on the timeline updates other of 0026].

As per claim 40, Deutscher teaches: one or more computer readable media storing computer executable instructions that, when executed by a computer, direct the computer to perform the method above [computer readable media of 0073].

As per claim 49, Deutscher teaches: rendering a first said node to output a referenced first said media [rendering of 0027-0028 of tree and nodes of 0011, 0023, 0163-0166 and Figure 19 for the output of a presentation]; during the rendering, changing one or more properties of a second said node; and initiating, by an event generator located on the second said node, an event for communication to a parent said node of the second said node, wherein the event describes the changing [changes to an event on the timeline updates others of 0026].

As per claim 50, Deutscher teaches: the event is communicated to at least one of an application over the API [changes to an event on the timeline updates other of 0026] and a timeline source for rendering the media timeline [rendering presentation timeline of 0027-0028].

As per claim 51, Deutscher teaches: one of the properties is node changed event [edit properties for change events of 0201].

As per claim 52, Deutscher teaches: one or more nodes is configured as a root node that specifies a starting point for rendering the media timeline [root node of Figure 19].

As per claim 53, Lamkin further teaches: the media timeline is configured for dynamic loading such that metadata included in at least one said node specifies a collection of nodes to be loaded when the media timeline is rendered [intelligent loading of 0387].

As per claim 55, Deutscher teaches: at least one said node is configured for communication of events to another said node such that a change may be made to the media timeline while the media timeline is rendered [changes to an event on the timeline updates other of 0026].

As per claim 56, Deutscher teaches: at least one node is configured to reference an effect to be applied to an output of media referenced by the node [script commands and events of 0018].

As per claim 57, Deutscher teaches: one or more computer readable media storing computer executable instructions that, when executed by a computer, direct the

computer to perform the method above [computer readable media of 0073].

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

Claims 1-2, 4-5, 7-9, 11-18, 20-26, 28, 30-32, 41, 44-45, 47-48, 54, 58-62, 64-68, 70-79, 82-87 and 89-90 rejected under 35 U.S.C. 103(a) as being unpatentable over

Deutscher (US 2004/0001106), and further in view of Lamkin (US 2004/0220926).

As per claim 1, Deutscher teaches a method comprising: receiving a request, from an application at an application programming interface (API), to interact with a plurality of media [multimedia presentation production system and process of 0007]; and generating a media timeline based on the request, wherein the media timeline [timeline of 0011 and 0026]: is exposed to the application via the API [timeline of 0011 and 0026]; includes a plurality of nodes and defines a presentation, to be output via one or more computers, of a first said media referenced by a first node with respect to a second media referenced by a second node [tree and nodes of 0011, 0023, 0163-0166 and Figure 19]; wherein: the first and second nodes are configured as parallel nodes such that the first node that is a child of a parent node is rendered concurrently with the second node that is a child of the same parent node [tree and nodes of 0011, 0023, 0163-0166 and Figure 19]; and at least one node includes metadata [one of the tree elements is a master track file that contains indexing, timing and track metadata of 001], the metadata describing: rendering of the at least one node [timing metadata of 0011]; and a collection of additional nodes to be dynamically modified when the media timeline is rendered [metadata for dynamic changes to output during rendering of 0213-0221 and 0230], but fails to expressly disclose the media timeline is configured for dynamic creation such that at least one node is created while the media timeline is being rendered, as recited in the claims. In the same field of the invention, Lamkin discloses loading a media collection and collection metadata for the playback of media, where the stream of media can have predefined jump points in the entity metadata to instruct the

playback system to intelligently load the stream (start loading at multiple points in the stream to enable quick jumping). Further, some predictive analysis is optionally used by the playback system [0387]. It would have been obvious to one of ordinary skill in the art, having the teachings of Deutscher and Lamkin before him at the time the invention was made, to modify metadata for loading of nodes in a presentation taught by Deutscher to include intelligently loading of Lamkin, in order to obtain a dynamic creation of nodes in a presentation. One would have been motivated to make such a combination to skip loading a part of the collection that is not wanted by the user, as taught by Lamkin [0387].

As per claim 2, Deutscher teaches: one or more nodes are configured as a sequence node such that one said node that is a child of the sequence node is rendered after another said node that is also a child of the sequence node [Fig 23].

As per claim 4, Deutscher teaches: one or more nodes is configured as a root node that specifies a starting point for rendering the media timeline [root node of Figure 19].

As per claim 5, Deutscher teaches: the first and second nodes reference the respective first and second media utilizing respective first and second pointers [pointers to media of 0007].

As per claim 7, Deutscher teaches: metadata is a start time property that specifies when rendering of the at least one said node is to begin with respect to another said node [metadata of 0213-0221 and 0230].

As per claim 8, Deutscher teaches: at least one node is configured to reference

an effect to be applied to an output of media referenced by the node [script commands and events of 0018].

As per claim 9, Lamkin further teaches: the media timeline is configured for dynamic loading such that metadata included in at least one said node specifies a collection of nodes to be loaded when the media timeline is rendered [intelligent loading of 0387].

As per claim 11, Deutscher teaches: at least one node is specified as read-only [cannot be edited of 0017].

As per claim 12, Deutscher teaches: at least one said node is configured for communication of events to another said node such that a change may be made to the media timeline while the media timeline is rendered [changes to an event on the timeline updates other of 0026].

As per claim 13, Deutscher teaches: the first and second said media have different formats [formats of 0007].

As per claim 14, Deutscher teaches: one or more computer readable media storing computer executable instructions that, when executed by a computer, direct the computer to perform the method above [computer readable media of 0073].

As per claim 15, Deutscher teaches a method comprising: generating a media timeline by an application, wherein the media timeline: includes a plurality of nodes [timeline of 0011 and 0026]; and defines a presentation of a first said media referenced by a first said node with respect to a second said media referenced by a second said node, the presentation being configured to be output by one or more computers [tree

and nodes of 0011, 0023, 0163-0166 and Figure 19]; and passing the media timeline to a timeline source for rendering [rendering of 0027-0028], but fails to expressly disclose dynamic creation such that at least a first node grouping is created while a second node grouping in the media timeline is being rendered, as recited in the claims. In the same field of the invention, Lamkin discloses loading a media collection and collection metadata for the playback of media, where the stream of media can have predefined jump points in the entity metadata to instruct the playback system to intelligently load the stream (start loading at multiple points in the stream to enable quick jumping). Further, some predictive analysis is optionally used by the playback system [0387]. It would have been obvious to one of ordinary skill in the art, having the teachings of Deutscher and Lamkin before him at the time the invention was made, to modify metadata for loading of nodes in a presentation taught by Deutscher to include intelligently loading of Lamkin, in order to obtain a dynamic creation of nodes in a presentation. One would have been motivated to make such a combination to skip loading a part of the collection that is not wanted by the user, as taught by Lamkin [0387].

As per claim 16, the claim contains substantially the same subject matter as claim 13, and remains rejected using the same rationale.

As per claim 17, the claim contains substantially the same subject matter as claim 8, and remains rejected using the same rationale.

As per claim 18, the claim contains substantially the same subject matter as claim 9, and remains rejected using the same rationale.

As per claim 20, the claim contains substantially the same subject matter as claim 14, and remains rejected using the same rationale.

As per claim 21, Deutscher teaches a method for outputting a media presentation via one or more computers comprising: specifying an effect to be applied to one or more of a plurality of media when the media is rendered [script commands and events of 0018]; and generating a media timeline configured for exposure via an application programming interface (API) [timeline of 0011 and 0026], wherein: the media timeline includes a plurality of nodes [tree and nodes of 0011, 0023, 0163-0166 and Figure 19]; two or more said nodes reference respective said media [tree and nodes of 0011, 0023, 0163-0166 and Figure 19]; one or more said nodes that reference the one or more said media include metadata that describes the effect [metadata for dynamic changes to output during rendering of 0213-0221 and 0230]; and at least one node includes metadata [one of the tree elements is a master track file that contains indexing, timing and track metadata of 001], the metadata describing: rendering of the at least one node [timing metadata of 0011]; and a collection of additional nodes to be dynamically modified when the media timeline is rendered [metadata for dynamic changes to output during rendering of 0213-0221 and 0230], but fails to expressly disclose the media timeline is configured for dynamic creation such that at least one node is created while the media timeline is being rendered, as recited in the claims. In the same field of the invention, Lamkin discloses loading a media collection and collection metadata for the playback of media, where the stream of media can have predefined jump points in the entity metadata to instruct the playback system to intelligently load the stream (start

loading at multiple points in the stream to enable quick jumping). Further, some predictive analysis is optionally used by the playback system [0387]. It would have been obvious to one of ordinary skill in the art, having the teachings of Deutscher and Lamkin before him at the time the invention was made, to modify metadata for loading of nodes in a presentation taught by Deutscher to include intelligently loading of Lamkin, in order to obtain a dynamic creation of nodes in a presentation. One would have been motivated to make such a combination to skip loading a part of the collection that is not wanted by the user, as taught by Lamkin [0387].

As per claim 22, Deutscher teaches: the effect is a simple effect provided by a software component that is configured to: receive a single stream of media; apply the effect to the single stream; and output the applied single stream [script commands and events of 0018].

As per claim 23, Deutscher teaches: the effect is a composite effect provided by a software component that is configured to: receive at least two streams of media; apply the effect at least two streams; and output a applied single stream of media composed of the applied at least two streams [script commands and events of 0018].

As per claim 24, Deutscher teaches: the effect is a composite effect provided by a software component that is configured to analyze at least two streams of media or output at least two streams of media [script commands and events of 0018].

As per claim 25, Deutscher teaches: the effect is a transition effect to be applied as a transition from a first media referenced by a first said node to a second media referenced by a second node [script commands and events of 0018].

As per claim 26, the claim contains substantially the same subject matter as claim 7, and remains rejected using the same rationale.

As per claim 28, the claim contains substantially the same subject matter as claim 9, and remains rejected using the same rationale.

As per claim 30, the claim contains substantially the same subject matter as claim 11, and remains rejected using the same rationale.

As per claim 31, the claim contains substantially the same subject matter as claim 12, and remains rejected using the same rationale.

As per claim 32, the claim contains substantially the same subject matter as claim 14, and remains rejected using the same rationale.

As per claim 41, Deutscher teaches in a media timeline exposed via an application programming interface, the media timeline having a plurality of nodes, at least two of which reference respective media, one or more nodes each having metadata that references a node grouping, a method comprising: utilizing a computer to load a first said node for rendering [rendering of 0027-0028]; examining metadata associated with the first node to determine a first said node grouping to be loaded in conjunction with the first node [metadata of 0213-0221 and 0230]; loading each said node referenced by the first node grouping; rendering the first node grouping [rendering of 0027-0028 of tree and nodes of 0011, 0023, 0163-0166 and Figure 19]; examining a second node in the first node grouping to determine a second said node grouping, wherein the examining at least one node in the first node grouping is performed during the rendering of the first node grouping [rendering presentation timeline of 0027-0028];

loading each node referenced by the second node grouping [rendering of 0027-0028 of tree and nodes of 0011, 0023, 0163-0166 and Figure 19]; and rendering the second node grouping when the rendering of the first said node grouping is completed [rendering of 0027-0028 of tree and nodes of 0011, 0023, 0163-0166 and Figure 19], wherein: and at least a fourth node is configured for communication of an initiated event to a fifth node which has subscribed to receive events initiated by the fourth node, such that a change is made to one or more nodes in the media timeline that are affected by the initiated event [changes to an event on the timeline updates other of 0026], wherein the one or more nodes of the media timeline that are affected by the initiated event are dynamically updated [changes to an event on the timeline updates other of 0026], but fails to expressly disclose the media timeline is configured for dynamic creation where at least a third node is created while the media timeline is being rendered, the dynamic creation of the third node being performed by a node source that includes data that defines properties and interrelationships of the created third node with respect to one or more nodes in the first node grouping or one or more nodes in the second node grouping, as recited in the claims. In the same field of the invention, Lamkin discloses loading a media collection and collection metadata for the playback of media, where the stream of media can have predefined jump points in the entity metadata to instruct the playback system to intelligently load the stream (start loading at multiple points in the stream to enable quick jumping). Further, some predictive analysis is optionally used by the playback system [0387]. It would have been obvious to one of ordinary skill in the art, having the teachings of Deutscher and Lamkin before him at the time the invention

was made, to modify metadata for loading of nodes in a presentation taught by Deutscher to include intelligently loading of Lamkin, in order to obtain a dynamic creation of nodes in a presentation. One would have been motivated to make such a combination to skip loading a part of the collection that is not wanted by the user, as taught by Lamkin [0387].

As per claim 44, the claim contains substantially the same subject matter as claim 8, and remains rejected using the same rationale.

As per claim 45, the claim contains substantially the same subject matter as claim 11, and remains rejected using the same rationale.

As per claim 47, the claim contains substantially the same subject matter as claim 13, and remains rejected using the same rationale.

As per claim 48, the claim contains substantially the same subject matter as claim 14, and remains rejected using the same rationale.

As per claim 54, Deutscher teaches a media timeline but fails to expressly disclose the media timeline is configured for dynamic creation such that at least one node is created while the media timeline is being rendered, as recited in the claims. In the same field of the invention, Lamkin discloses loading a media collection and collection metadata for the playback of media, where the stream of media can have predefined jump points in the entity metadata to instruct the playback system to intelligently load the stream (start loading at multiple points in the stream to enable quick jumping). Further, some predictive analysis is optionally used by the playback system [0387]. It would have been obvious to one of ordinary skill in the art, having the

teachings of Deutscher and Lamkin before him at the time the invention was made, to modify metadata for loading of nodes in a presentation taught by Deutscher to include intelligently loading of Lamkin, in order to obtain a dynamic creation of nodes in a presentation. One would have been motivated to make such a combination to skip loading a part of the collection that is not wanted by the user, as taught by Lamkin [0387].

As per claim 58, Deutscher teaches an application programming interface embodied on a computer storage medium, which when interfaced with a computer, exposes a media timeline to one or more independent applications, the application programming interface comprising: each said node includes metadata that describes the node [metadata of 0213-0221 and 0230]; one or more said nodes reference a corresponding media item [tree and nodes of 0011, 0023, 0163-0166 and Figure 19]; the plurality of nodes are arranged in a tree structure [tree and nodes of 0011, 0023, 0163-0166 and Figure 19]; and the arrangement of the plurality of nodes, one to another, describes an order for rendering the plurality of nodes [tree and nodes of 0011, 0023, 0163-0166 and Figure 19; timeline of 0011 and Figure 23], but fails to expressly disclose wherein the media timeline is configured for dynamic creation such that at least one node is created while the media timeline is rendered and at least one node is dynamically updated in response to the at least one node being created, as recited in the claims. In the same field of the invention, Lamkin discloses loading a media collection and collection metadata for the playback of media, where the stream of media can have predefined jump points in the entity metadata to instruct the playback system

to intelligently load the stream (start loading at multiple points in the stream to enable quick jumping). Further, some predictive analysis is optionally used by the playback system [0387]. It would have been obvious to one of ordinary skill in the art, having the teachings of Deutscher and Lamkin before him at the time the invention was made, to modify metadata for loading of nodes in a presentation taught by Deutscher to include intelligently loading of Lamkin, in order to obtain a dynamic creation of nodes in a presentation. One would have been motivated to make such a combination to skip loading a part of the collection that is not wanted by the user, as taught by Lamkin [0387].

As per claim 59, the claim contains substantially the same subject matter as claim 7, and remains rejected using the same rationale.

As per claim 60, the claim contains substantially the same subject matter as claim 8, and remains rejected using the same rationale.

As per claim 61, the claim contains substantially the same subject matter as claim 6, and remains rejected using the same rationale.

As per claim 62, the claim contains substantially the same subject matter as claim 9, and remains rejected using the same rationale.

As per claim 64, the claim contains substantially the same subject matter as claim 11, and remains rejected using the same rationale.

As per claim 65, the claim contains substantially the same subject matter as claim 12, and remains rejected using the same rationale.

As per claim 66, Deutscher teaches: two or more said nodes reference

respective media; the plurality of nodes are arranged in a hierarchy to include a parent said node and a child said node [tree and nodes of 0011, 0023, 0163-0166 and Figure 19; timeline of 0011 and Figure 23]; and the child said node is configured for initiating an event for communication to the parent said node, wherein the event: is configured such that a change may be made to one or more properties of the child node while the media timeline is rendered and describes the change such that additional nodes associated with the child node are dynamically updated in accordance with the communicated event [changes to an event on the timeline updates other of 0026].

As per claim 67, Deutscher teaches: wherein another node, which is not a parent of the child node, subscribes to the child node to receive the event [changes to an event on the timeline updates other of 0026].

As per claim 68, Deutscher teaches: the event is initiated by the child node; and one or more events initiated by children of the child node [changes to an event on the timeline updates other of 0026].

As per claim 70, the claim contains substantially the same subject matter as claim 51, and remains rejected using the same rationale.

As per claim 71, the claim contains substantially the same subject matter as claim 4, and remains rejected using the same rationale.

As per claim 72, the claim contains substantially the same subject matter as claim 9, and remains rejected using the same rationale.

As per claim 73, the claim contains substantially the same subject matter as claim 10, and remains rejected using the same rationale.

As per claim 74, the claim contains substantially the same subject matter as claim 8, and remains rejected using the same rationale.

As per claim 75, the claim contains substantially the same subject matter as claim 11, and remains rejected using the same rationale.

As per claim 76, Deutscher teaches: exposing a media timeline comprising one or more nodes to the application [tree and nodes of 0011, 0023, 0163-0166 and Figure 19; timeline of 0011 and Figure 23]; enabling the application to call any of the one or more nodes [call for items of 0100 of timeline of 0011 and 0026], wherein each of the one or more nodes: references corresponding media [media of 0007]; includes metadata describing one or more properties for rendering corresponding media [metadata of 0213-0221 and 0230]; and includes metadata specifying the node as read-only [input and output nodes are not changed of 0081] but fails to expressly disclose configuring the media timeline for dynamic creation such that at least one of the one or more nodes is created while the media timeline is being rendered, as recited in the claims. In the same field of the invention, Lamkin discloses loading a media collection and collection metadata for the playback of media, where the stream of media can have predefined jump points in the entity metadata to instruct the playback system to intelligently load the stream (start loading at multiple points in the stream to enable quick jumping). Further, some predictive analysis is optionally used by the playback system [0387]. It would have been obvious to one of ordinary skill in the art, having the teachings of Deutscher and Lamkin before him at the time the invention was made, to modify metadata for loading of nodes in a presentation taught by Deutscher to include

intelligently loading of Lamkin, in order to obtain a dynamic creation of nodes in a presentation. One would have been motivated to make such a combination to skip loading a part of the collection that is not wanted by the user, as taught by Lamkin [0387].

As per claim 77, Deutscher teaches a system comprising: a plurality of media [media of 0011]; a plurality of applications [application programs of 0074]; and an infrastructure layer that: provides an application programming interface (API) for interaction by the plurality of applications with the plurality of media when any said application is executed [application programming interface of 0040]; and exposes a media timeline [timeline of 0011 and Figure 23], callable by the plurality of applications via the API upon an execution thereof [call of 0100], and that defines a presentation of the plurality of media [timeline presentation of 0011], wherein the media timeline: includes a plurality of nodes that each reference respective media [tree and nodes of 0011, 0023, 0163-0166 and Figure 19], but fails to expressly disclose wherein the media timeline is configured for dynamic creation such that at least one node is created while the media timeline is rendered and is configured for dynamic loading such that metadata included in the at least one node created specifies a collection of nodes to be loaded when the media timeline is rendered, as recited in the claims. In the same field of the invention, Lamkin discloses loading a media collection and collection metadata for the playback of media, where the stream of media can have predefined jump points in the entity metadata to instruct the playback system to intelligently load the stream (start loading at multiple points in the stream to enable quick jumping). Further, some

predictive analysis is optionally used by the playback system [0387]. It would have been obvious to one of ordinary skill in the art, having the teachings of Deutscher and Lamkin before him at the time the invention was made, to modify metadata for loading of nodes in a presentation taught by Deutscher to include intelligently loading of Lamkin, in order to obtain a dynamic creation of nodes in a presentation. One would have been motivated to make such a combination to skip loading a part of the collection that is not wanted by the user, as taught by Lamkin [0387].

As per claim 78, the claim contains substantially the same subject matter as claim 8, and remains rejected using the same rationale.

As per claim 79, Deutscher teaches: the media timeline defines a presentation of a first media referenced by a first node with respect to a second media referenced by a second node [tree and nodes of 0011, 0023, 0163-0166 and Figure 19; timeline of 0011 and Figure 23]; and at least one said node includes metadata that describes rendering of the at least one node [metadata of 0213-0221 and 0230].

As per claim 82, Deutscher teaches: at least one node is specified as read-only [cannot be edited of 0017].

As per claim 83, Deutscher teaches: at least one said node is configured for communication of events to another said node such that a change may be made to the media timeline while the media timeline is rendered [changes to an event on the timeline updates other of 0026].

As per claim 84, Deutscher teaches a computer comprising: a processor and memory configured to maintain: a plurality of media [media of 0011]; a plurality of

applications [application programs of 0074]; wherein each said application is configured to request at least one of editing, encoding, and rendering of the plurality of media [rendering of 0027-0028]; and an infrastructure layer configured to: provide an application programming interface (API) for interaction by the plurality of applications with the plurality of media [application programming interface of 0040]; and expose a media timeline [timeline of 0011 and Figure 23], callable by the plurality of applications via the API upon an execution thereof [call of 0100], and that defines a presentation of the plurality of media [timeline presentation of 0011], wherein the metadata describes: initiating rendering of the plurality of nodes [metadata of 0213-0221 and 0230]; properties and interrelationships of the plurality of nodes [metadata of 0213-0221 and 0230]; and node types of the plurality of nodes [metadata of 0213-0221 and 0230]; and dynamic change to the media timeline such that a group of nodes affected by a modification to an associated node are automatically updated in accordance with the modification as specified in the properties and interrelationships of the plurality of nodes [metadata for dynamic changes to output during rendering of 0213-0221 and 0230]; at least one node that is configured for communication of events to another node that is configured for communication of events to another node such that a change may be made to the media timeline while the media timeline is being rendered [changes to an event on the timeline updates other of 0026]; and at least one node that is a parallel node that provides simultaneous rendering of at least two child nodes [tree and nodes of 0011, 0023, 0163-0166 and Figure 19; timeline of 0011 and Figure 23], but fails to expressly disclose wherein the media timeline specifies delayed creation of one or more

said nodes when the media timeline is rendered, as recited in the claims. In the same field of the invention, Lamkin discloses loading a media collection and collection metadata for the playback of media, where the stream of media can have predefined jump points in the entity metadata to instruct the playback system to intelligently load the stream (start loading at multiple points in the stream to enable quick jumping). Further, some predictive analysis is optionally used by the playback system [0387]. It would have been obvious to one of ordinary skill in the art, having the teachings of Deutscher and Lamkin before him at the time the invention was made, to modify metadata for loading of nodes in a presentation taught by Deutscher to include intelligently loading of Lamkin, in order to obtain a dynamic creation of nodes in a presentation. One would have been motivated to make such a combination to skip loading a part of the collection that is not wanted by the user, as taught by Lamkin [0387].

As per claim 86, the claim contains substantially the same subject matter as claim 9, and remains rejected using the same rationale.

As per claim 87, the claim contains substantially the same subject matter as claim 8, and remains rejected using the same rationale.

As per claim 88, the claim contains substantially the same subject matter as claim 12, and remains rejected using the same rationale.

As per claims 89 and 90, Deutscher teaches: a root node that specifies a starting point for rendering the media timeline [tree and nodes of 0011, 0023, 0163-0166 and Figure 19; timeline of 0011 and Figure 23], the root node including metadata that

describes how rendering is to be initiated [one of the tree elements is a master track file that contains indexing, timing and track metadata of 001]; a leaf node that directly maps to media to be rendered and output, the leaf node including metadata that describes how rendering is supposed to be initiated; a sequence node that includes metadata that describes a rendering order of a plurality of leaf nodes to the sequence node; and a parallel node that includes metadata specifying a plurality of leaf nodes that are rendered simultaneously [tree and nodes of 0011, 0023, 0163-0166 and Figure 19; timeline of 0011 and Figure 23].

Response to Arguments

Applicant's arguments filed 7/2/2009 have been fully considered but they are not persuasive.

On pages 25-26 of the response, Applicant argues that Deutscher fails to disclose or suggest *metadata included in at least one node specifies a collection of nodes to be modified when the at least one node is loaded*, as recited in claim 33. Examiner respectfully disagrees. In the previous and current rejection, Examiner cited the metadata in paragraphs 0214 and 0233. Explanations of this metadata and how it relates to the modification of a collection of nodes during loading and/or rendering can at least be found in paragraphs 0014-0016. Numerous other examples of pertinent metadata exist throughout the specification, contrary to Applicant's statement on page 26 that no description that suggests metadata as recited in claim 33 exists in the Deutscher reference. Paragraphs 0014-0016 teach tools and grids for modifying metadata, attributes, properties, and other information about information that affect a

template, media file, such as an audio or video file and/or a master track media file displayed on a timeline. Some of the information that can be modified includes colors, fonts, layout features, export options, playback options, languages, dimensions settings and quality settings, among others. For example, as specified in Deutscher, a change of the language or quality setting of a master track file will affect all events in the presentation, i.e. all nodes, thus *specifying a collection a collection of nodes to be modified when the at least one node is loaded.*

On pages 26-27 of the response, Applicant provides similar arguments for claim 49. Specifically, Applicant argues that Deutscher fails to teach during the rendering, changing one or more properties of a second node, as recited in claim 49. Examiner respectfully disagrees. As stated above, paragraphs 0014-0016 teach tools and grids for modifying metadata, attributes, properties, and other information about information that affect a template, media file, such as an audio or video file and/or a master track media file displayed/rendered on a timeline. Some of the information that can be modified includes colors, fonts, layout features, export options, playback options, languages, dimensions settings and quality settings, among others. For example, as specified in Deutscher, a change of the language or quality setting of a master track file will affect all events in the presentation, i.e. all nodes, thus *changing one or more properties of a second node during the rendering.*

On pages 27-28 of the response, Applicant argues that the Deutscher reference (U.S. Pub 2004/0001106) should be disqualified under 103(c) because the reference only qualifies as prior art under 102(e). Examiner respectfully disagrees. Deutscher

has a publication date of 1/1/2004, while the current application has a filing date of 2/19/2004. Thus, the Deutscher reference is available and applied as a 102(a) reference.

Conclusion

THIS ACTION IS MADE FINAL. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to ENRIQUE W. ITURRALDE whose telephone number is (571)270-3627. The examiner can normally be reached on Monday-Thursday 9 AM - 5 AM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Weilun Lo can be reached on (571)272-4847. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/E. W. I./
Examiner, Art Unit 2179

/Ba Huynh/

Primary Examiner, Art Unit 2179